Patrick Freeman

N8890862

INB383

# Statement of Completeness

Things I didn’t do

# Path Finding

**A\* Search Algorithm**

* Uses a heuristic to estimate the cost to the goal
* Guaranteed to find the shortest path (admissible)
* Can take a long time on large graphs
* Good when an absolute result is needed

**Greedy Search Algorithm**

* Uses a heuristic to find the cost of choosing a node
* Uses the cheapest option first (best-first-search)
* Doesn’t always find the shortest overall path
* Very Fast

# Finite State Machines

**Bold** = Default state

### Enemy1

States: patrol (), **stop (),** crazy ()

### Enemy2 / Enemy3

States: **Patrol (),** Alarm ()

### Enemy4

States**: Patrol()**, Attack ()

### Doors

States: Open (), **Closed ()**

## Triggers

### Trigger A

**Door1:** Open ()/Closed ()

**Enemy1**: patrol ()

### Trigger B

**Door2:** Open ()/Closed ()

### Trigger C

**Door3:** Open ()

**Enemy1**: Crazy ()

### Trigger D

**Enemy4**: attack ()

### Trigger E

Door4: open ()

### Trigger F

Enemy2/Enemy3: Alarm ()

1. State Diagram
2. Trigger layout

# Conclusion